

УДК 004.054

ОСОБЛИВОСТІ РОЗРОБКИ МОБІЛЬНИХ ІГРОВИХ ДОДАТКІВ НА ПЛАТФОРМАХ UNITY ТА CONSTRUCT 2

Єлькін Валентин

Науковий керівник канд. ф-м. наук, доцент Паращук С.Д.

Центральноукраїнський державний педагогічний університет імені

Володимира Винниченка, м. Кропивницький, Україна

В статті розглядаються особливості розробки мобільних ігрових додатків на платформах Unity та Construct 2. Проаналізовано переваги та особливості кожної із платформ та потенціальні можливості їх використання. В якості мобільного ігрового додатку обрано ігровий додаток типу breakout. Проаналізовано особливості роботи Unity та Construct 2 з фізикою та логікою гри на обох платформах і зроблені певні висновки про потенційні можливості кожного інструменту.

Ключові слова: мобільні додатки, ігровий рушій, Unity, Construct 2, browser.

FEATURES OF DEVELOPMENT OF MOBILE GAME ADDITIONS ON PLATFORM UNITY AND CONSTRUCT 2

Elkin Valentin

Scientific supervisor: Candidate of Sciences (Physics and Mathematics),

Docent Parashuk S.D.

The Volodymyr Vynnychenko Central Ukrainian State Pedagogical University,

Kropyvnytsky, Ukraine

The article discusses the features of developing mobile gaming applications on the Unity and Construct 2. platforms. As a mobile gaming application, a game application of the breakout type is selected. The features of Unity and Construct 2 with the physics and logic of the game on both platforms are analyzed and some conclusions are drawn about the potential capabilities of each tool.

Keywords: mobile applications, game engine, Unity, Construct 2, browser.

Постановка проблеми. Ігрова індустрія на даний момент розвивається дуже швидко. Існує велика кількість ігрових компаній, які розробляють колосальні проекти. Основну роль в розробці ігор відіграють програмісти. Вони займаються створенням базових механік гри: від них залежить, те як буде працювати певна механіка гри та на скільки коректно буде працювати гра з

точки зору реальної фізики. Гарний програміст має добре знати не тільки основні мови програмування, а й вміти використовувати математику, фізику, вміти швидко пристосовуватися до нового.

Аналіз досліджень та публікацій.

Актуальною для розробки ігрових додатків є Android - операційна система, яка працює на пристроях з операційною системою Android. Операційну систему android розробляє компанія Google. Розробка мобільних ігрових додатків має низку послідовних етапів. Але щоб створити мобільний додаток потрібен певний інструмент, особливо якщо цей додаток ігровий. На практиці використовуються рушії, де вже реалізована велика кількість потрібних функцій для створення ігрових додатків. Дослідження деяких рушіїв та перспектив розробки мобільних додатків на android можна знайти в роботах Авраменко В.А., Гусейнова У.Ю., Регета Д. М., Неймана А.А., Куцаченко Н.Г., та в багатьох інших роботах.

Мета статті. Метою статті є аналіз застосування найбільш популярних інструментів для реалізації ігрових android додатків на прикладі розробки гри типу breakout.

Виклад основного матеріалу дослідження.

Візуалізатор Unity

Unity це багатоплатформовий інструмент для розробки 2d та 3d ігрових додатків які працюють на операційних системах Android і iOS. В ньому реалізована велика кількість інструментів для створення інтернет-додатків в додавання спеціальних під'єднаних модулів для браузера в самому Unity. Додатки створені за допомогою Unity, підтримують DirectX та OpenGL [4].

В середину Unity влаштований не складний редактор Drag & Drop інтерфейс, з який не складно налаштовується, його функціонал включає велику кількість різних комбінацій екранів, за допомогою таких можливостей і можна робити такі тонкі налаштування, всього проекту додатку одразу в середині редактора Unity. В Unity є підтримка таких мов кодування як: C#, JavaScript (не оригінальна). Розробка кожного проекту в Unity з розробки незалежних сцен та

окремих файлів, які знаходяться в середині себе, а також в самому рушії є свій магазин в якому можна знайти багато різних об'єктів, анімації, ефектів для цих анімацій та ще велику кількість налаштувань для проекту (рис 1).

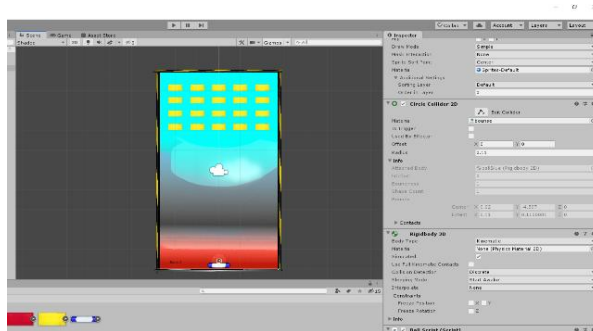


Рис. 1 Вікно створення фізичних можливостей

В кожній сцені міститься багато об'єктів (моделі), а пусті об'єкти – це ті об'єкти що це не мають жодної ігрової моделі. Кожен об'єкт має в собі набори деяких компонентів, з якими вони будуть взаємодіяти скрипти. Кожен такий елемент має свою назву (Unity надає можливість називати різні об'єкти схожими назвами), вони мають свої теги та шари, на яких вони будуть зображатися [1].

Логіка гри задається за допомогою написання певних скриптів до кожних елементів. Та додаванню цих скриптів до потрібного об'єкта. Кожен скрипт має два стартових методи, start та update. В них описуються основні дії, та явища які будуть відбуватися з періодичним викликом.

Construct 2

Construct 2 надає можливість всім хто хоче створювати 2D-ігри незалежно від її складності та ігрового жанру. В ньому можна створювати ігри без написання коду гри. Ігрові додатки, розроблені в ньому, дуже легко переносяться на операційні системи - PC, Mac, Linux, браузері з підтримкою HTML5, Android, iOS, Windows Phone, та ін. Підтримка Android і iOS виконується за допомогою технології Cocos2D, directCanvas і Intel XDK.

Рушій має інтуїтивно зрозумілий інтерфейс, а також не потребує попереднього досвіду розробки ігрових додатків, тобто він має низький поріг входження. Логіка ігрових додатків в Construct 2 задається за допомогою системи подій (Events) які пов'язані з ними[5]. Даний ігровий рушій підходить

як для початківців, які хочуть спробувати щось створити для себе, так і для профі. Рушій Construct 2 написано на мові C ++, кодування додатку відбувається на Javascript (рис 2). Під час експорту проекту JavaScript змінюється, але можна приєднати JS Plugin SDK та змінювати код власноруч. Перевагою Construct 2 є те, що він має модульний дизайн, тому якщо якийсь модуль не буде задіяно, то під час збірки проекту він не буде використаний і розмір файлу буде меншим.

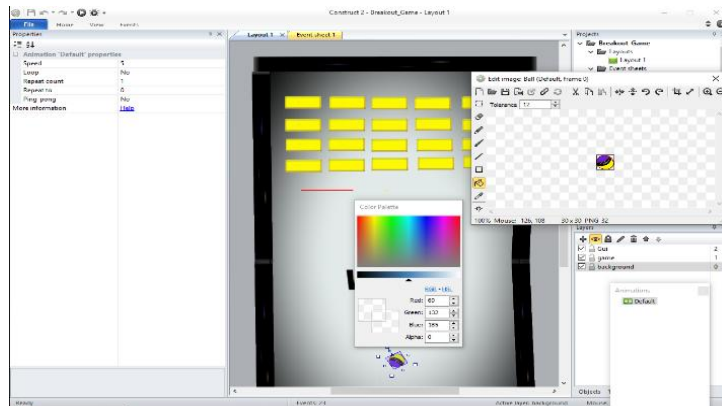


Рис. 2 Редактор в Construct 2

Також існує багато плагінів, які розширюють початкові можливості Construct 2. Для перевірки роботи щойно створеного додатка, його одразу можна запустити кнопкою Build and run, Після цього проект відкриється в браузері який вибраний по замовчуванню у користувацькій системі. Якщо розробку закінчено тоді можна експортувати створений проект в ту систему яка потрібна.

Отже, ми розглянули два найбільш популярних на даний час сучасних засоби для розробки мобільних додатків. Далі розглянемо процес створення android додатку на Construct 2 та Unity. Невеликим мінусом є орієнтованість тільки на 2D розробку в Construct 2, тому додаток будемо розробляти у 2D режимі.

Для дослідження можливостей обох платформ було розроблено гру, подібну до аркадної гри breakout. В даній грі є ігрова ракетка що знаходиться в нижній частині екрану, якою може керувати гравець. Блоки які виступають в ролі перешкод, вони знаходяться у верхній частині ігрового поля. Та куля яка

може знищувати перешкоди та відбиватись від ракетки. Кінець гри настає коли кулька знищує всі блоки, або вона доходить до границь нижнього поля і ракетка не може її відбити.

Розробка аркадної гри box destroyer на Construct 2.

Для розробки гри на конструкторі, в рамках нашого дослідження обрано популярний конструктор ігор Construct 2

Розробка графіки гри. Графіка створюється в редакторі що надається самим конструктором. Для розробки можна використовувати різні фільтри для зображення. Для початку потрібно додати на екран спрайт і перейти до редагування зображення. Додамо невелике зображення та додамо до нього 4 точки, їх буде достатньо для обробки колізій. Надалі будемо додавати інші властивості які допоможуть налаштувати коректну поведінку кульки.

Коли ми додали всі об'єкти, можна перейти до налаштування властивостей цих об'єктів. Треба розташувати всі елементи по своїх позиціях.

Робота з фізикою гри. Всі спрайти що були створені мають полігони які будуть відповідати границям зображення. Треба розуміти що їх можна завжди змінити як нам потрібно. Основним елементом буде обробка колізій. Вона буде відбуватися коли кулька буде зіштовхуватися з іншим об'єктом. Тобто їх координати будуть збігатися. Для границь екрану, щоб кулька не вилітала за нього було додано чотири блоки, від яких буде відбиватися кулька.

Робота над логікою гри.

У Construct 2 для створення фізики гри існує один із видів поведінки як Physics він працює на Box2DWeb. Він дозволяє переміщатися об'єктам із використанням явищ фізики. Для того що представляє кожна поведінка можна скористатися прикладами які доступні в конструкторі. Всі приклади, пов'язані з фізикою починаються зі слова Physics. Аби надати об'єктам фізичних властивостей, треба натиснути на спрайт, до якого хочемо додати фізику. Та у вікні Properties натиснути на Add / Edit under Behaviors, і натиснути на плюс, та обрати Physics.

Для всіх об'єктів що були додані діє гравітація, яка тягне їх до низу, сила гравітації дорівнює 0. Для того щоб її зміни, треба перейти на вкладку properties та знайти пункт set world gravity. Гравітація діє для всіх об'єктів що знаходяться на екрані однаково, якщо відключити гравітацію одного спрайту, тоді гравітація не буде діяти і на інші спрайти. Аби гравець міг керувати об'єктами треба додати обробник, який буде реагувати на натискання. В конструкторі вже є вбудовані такі обробники (рис 3).

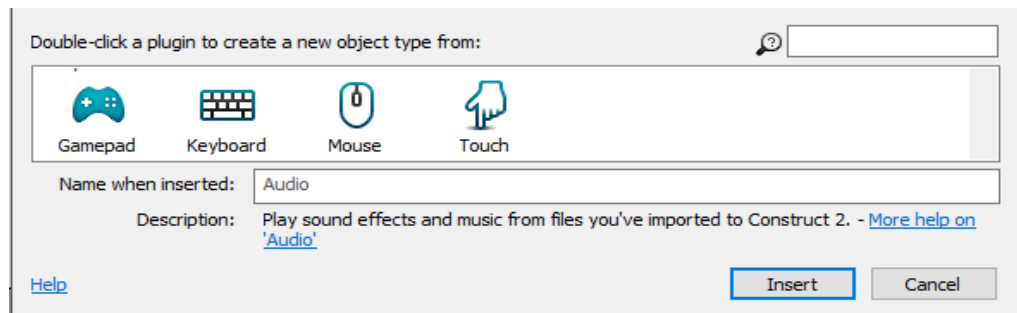


Рис. 3 Додавання обробника подій

За допомогою них відбувається поєднання подій з спрайтами. Робота з тим чи іншим об'єктом полягає побудові певної логіки їх використання на сцену на настроюванні його властивостей. Використовуючи ці об'єкти було додано до сцени бекграунд, контролер для натискання і обробки подій, блоки кульку та платформу та ігрові границі. Таким чином ми можемо ділі задавати нашу логіку гри.

Розробка аркадної гри box destroyer на Unity.

Для розробки гри на рушії, в рамках нашого дослідження, обрано Unity.

Робота з графікою. Для того щоб створити графіку треба скористатися будь яким графічним редактором і створити всі необхідні асети для проекту і додати в проект. Де вже сам рушій перетворить їх в зразок текстури тобто спрайт. Також є вбудований редактор спрайтів, в яких можна створити та нарізати їх на частини потрібні.

Робота з фізикою гри. Колайдери можуть бути додані до об'єкта без компонента жорсткого тіла для створення підлоги, стін та інших нерухомих елементів сцени. Вони відносяться до статичних колайдерів. Взагалі, нам не слід переставляти статичні колізії, змінюючи положення Transform, оскільки це

сильно вплине на працездатність фізичного двигуна. Колайдери на об'єкті, що мають жорстке тіло, відомі як динамічні колайдери. Статичні колайдери можуть взаємодіяти з динамічними колайдерами, але оскільки у них немає жорсткого тіла, вони не рухатимуться у відповідь на зіткнення (рис 4).

Для початку треба додати компонент Collider до об'єкту нашої гри, Collider буде означати тип обраного рушія, ми будемо використовувати Circle 2D Collider. Ми будемо колайдер використовувати для генерації зіткнень та надання ефекту відбиття (колізії) для нашої кульки [1].

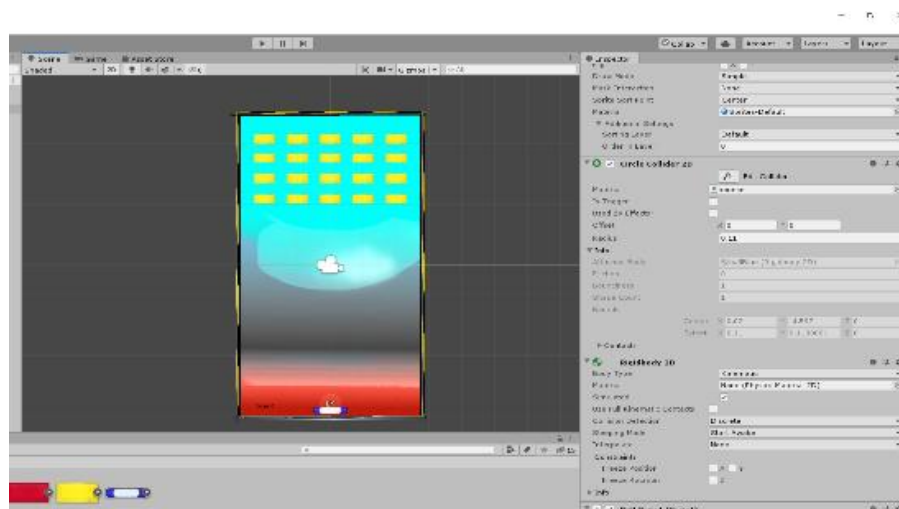


Рис. 4 Вікно створення фізичних можливостей

Однією з най важливіших деталей в нашому проєкті є налаштування колізії є (Layers) в них можна назначити для кожного потрібного об'єкта свій рівень доступу. Якщо об'єкти будуть знаходитись на різних рівнях доступу вони не можуть взаємодіяти між собою. Тому ми не будемо залишати увімкненою взаємодію кульки між платформою, та гравцем із блоками перешкодами.

Робота над логікою гри.

Логіка гри задається за допомогою написання певних скриптів до кожних елементів та додаванню цих скриптів до потрібного об'єкта. Кожен скрипт має два стартових методи це - старт та апдейт. В них ми будемо записувати основні дії та концепції та певні явища які будуть відбуватися з періодичним викликом. В результаті створений скрипт можна додати до багатьох об'єктів. Наприклад,

скрипт, що під час кожного виклику методу Update перевіряє скільки об'єктів знищив гравець та змінює кількість балів в певному полі.

Висновки та перспективи подальших пошуків у напрямі дослідження.

В статті були розглянуті особливості розробки ігрових додатків на двох інструментах - Unity та Construct 2. В якості об'єкта дослідження для аналізу складності реалізації основного функціоналу кожного інструменту розглядався ігровий додаток, схожий на гру Breakout.

Дослідження показало, що за потенційними можливостями та потужністю ігровий рушій Unity переважає конструктор Construct 2 та більше підходить для розробки великих та середніх проектів.

Разом з тим, на Construct 2 розробка була швидшою, він має зручний інтерфейс, та багато вбудованого функціонала що реалізує потрібні можливості, що робить конструктор ефективним інструментом для розробки невеликих простих ігор.

Список використаних джерел.

1. Документація по Android [Електронний ресурс]: Collider. – Режим доступу: <https://developer.android.com/docs>.
2. Документація по Unity [Електронний ресурс]: Unity 3D. – Режим доступу: <https://unity.com/ru>.
3. Колізії та їх застосування [Електронний ресурс]: Операційні системи. – Режим доступу: https://howlingpixel.com/iuk/Ambient_occlusion.
4. Операційні_системи [Електронний ресурс]: Операційні системи. – Режим доступу: https://wiki.cuspu.edu.ua/index.php/Операційні_системи.
5. Офіційний сайт Construct 2 [Електронний ресурс]: Construct 2. – Режим доступу: <https://www.scirra.com/>.
6. Android operation information [Електронний ресурс]: Android O S. – Режим доступу: <https://www.gadgetgyani.com/wp-content/uploads/2016/03/android-features-pdf.pdf>.